

Циклы *while* и *for*

Цикл `while`

что это такое?

- Задача цикла: выполнить код несколько раз подряд.
- Состав цикла:
 - заголовок: содержит слово `while` и условие выполнения цикла
 - тело: содержит полезную нагрузку
- Пример: цикл, который печатает числа от 1 до 5:

```
x = 1
while x <= 5:
    print(x)
    x = x + 1
```


Цикл `while`

ключевые слова

- `break`: завершить цикл аварийно
- `continue`: сразу перейти на следующую итерацию
- `else`: содержит код, который выполнится сразу после завершения цикла. Кроме случая если цикл завершается из-за `break`

```
x = 0
while x < 10:
    x += 1
    if x == 5:
        continue
    if x == 8:
        break
else:
    print("Цикл завершен")
print("Цикл точно завершен")
```

Цикл `while`

БУДЬТЕ ОСТОРОЖНЫ, жизнь не бесконечна!!!

- Если не уважать цикл `while` он убежит из-под контроля и попытается захватить мир!
- Это конечно же шутка 😊 - мир он не захватит, но ваш компьютер может
- Пример. Этот цикл будет бесконечно выводить все числа подряд:

```
x = 1
while x > 0:
    print(x)
    x = x + 1
```


Отличие `for` от `while`

они двойняшки))

- Цикл `while`:
Используется для повторения блока кода, пока выполняется определённое условие.
Количество итераций может быть неизвестно заранее и зависит от условия.
- Цикл `for`:
Используется для итерации по элементам последовательности (например, списки, строки, кортежи, диапазоны и т.д.).
Количество итераций обычно известно заранее (определяется длиной последовательности).

Пример while

```
• correct_password = "Secure123" # Правильный пароль
  max_attempts = 3                # Максимальное количество попыток
  attempts = 0                    # Счетчик попыток

print("Добро пожаловать в систему безопасного входа!")

while attempts < max_attempts:
    password = input("Введите ваш пароль: ")

    # Проверка, хочет ли пользователь выйти
    if password.lower() == "выход":
        print("Вы решили выйти. До свидания!")
        break # Выход из цикла, если пользователь выбрал выход

    # Проверка пароля
    if password == correct_password:
        print("Доступ разрешен! Добро пожаловать.")
        break # Выход из цикла, если пароль верный
    else:
        attempts += 1
        if attempts < max_attempts:
            print(f"Неверный пароль. У вас осталось {max_attempts - attempts} попыток.")
            continue # Пропустить оставшуюся часть цикла и запросить пароль снова
else:
    # Этот блок выполняется, если цикл завершился без 'break' (т.е. слишком много
    # неудачных попыток)
    print("Слишком много неудачных попыток. Ваш аккаунт заблокирован.")
```


Пример for

```
• correct_password = "Secure123" # Правильный пароль
  max_attempts = 3               # Максимальное количество попыток
  attempts = 0                   # Счетчик попыток

print("Добро пожаловать в систему безопасного входа!")

for element in [1,2,3,4,5]:
    password = input("Введите ваш пароль: ")

    # Проверка, хочет ли пользователь выйти
    if password.lower() == "выход":
        print("Вы решили выйти. До свидания!")
        break # Выход из цикла, если пользователь выбрал выход

    # Проверка пароля
    if password == correct_password:
        print("Доступ разрешен! Добро пожаловать.")
        break # Выход из цикла, если пароль верный
    else:
        attempts += 1
        if attempts < max_attempts:
            print(f"Неверный пароль. У вас осталось {max_attempts - attempts} попыток.")
            continue # Пропустить оставшуюся часть цикла и запросить пароль снова
else:
    # Этот блок выполняется, если цикл завершился без 'break' (т.е. слишком много
    неудачных попыток)
    print("Слишком много неудачных попыток. Ваш аккаунт заблокирован.")
```